LOAN DOCUMENT

PHOTOGRAPH THIS SHEET

DTIC ACCESSION NUMBER

LEVEL

INVENTORY

0

Functional Process Simulation

DOCUMENT IDENTIFICATION

Dec 93

DISTRIBUTION STATEMENT

ACCESSION FOR	
NTIS GRA&I	☑
DTIC TRAC	☐
UNANNOUNCED	☐
JUSTIFICATION	
BY	
DISTRIBUTION/	
AVAILABILITY CODES	
DISTRIBUTION	AVAILABILITY AND/OR SPECIAL
A-1	

DISTRIBUTION STAMP

DTIC ELECTE AUG 1 1995 S C D

DATE ACCESSIONED

HANDLE WITH CARE

DATE RETURNED

19950725 001

DATE RECEIVED IN DTIC

REGISTERED OR CERTIFIED NUMBER

PHOTOGRAPH THIS SHEET AND RETURN TO DTIC-FDAC

Functional Process Simulation

A Guidebook

(Second Edition)

December 1993

Prepared for:

Deputy Assistant Secretary of Defense
(Information Management)
Office of the Assistant Secretary of Defense
(Command, Control, Communications and Intelligence)
Washington, DC 20301-3040

DRAFT SF 298

1. Report Date (dd-mm-yy) DEC 1993	2. Report Type	3. Dates covered (from... to)
4. Title & subtitle FUNCTIONAL PROCESS SIMULATION A GUIDEBOOK (SECOND EDITION)		5a. Contract or Grant # 5b. Program Element #
6. Author(s)		5c. Project # 5d. Task # 5e. Work Unit #
7. Performing Organization Name & Address DEPUTY ASSISTANT SECRETARY OF DEFENSE (IM) OFFICE OF THE ASSISTANT SECRETARY OF DEFENSE (C3I) 1225 JEFFERSON DAVIS HGWY, STE 910 ARLINGTON, VA 22202-4301		8. Performing Organization Report #
9. Sponsoring/Monitoring Agency Name & Address SAME		10. Monitor Acronym 11. Monitor Report #

12. Distribution/Availability Statement DISTRIBUTION STATEMENT A; APPROVED FOR PUBLIC RELEASE, DISTRIBUTION UNLIMITED

13. Supplementary Notes **REPLACES THE GUIDEBOOK BY THE SAME TITLE, DATED 1/1/93, ADA260110.*** CIM (COLLECTION)

14. Abstract This Corporate Information Management (CIM) guidebook is part of a revolutionary program aimed at changing the way people work in the Department of Defense. Some of the methods associated with the Business Process Improvement Program (BPIP) include activity and data modeling. Using the developing system description standard, Integrated Computer Aided Definition (IDEF), this modeling helps professionals better understand their current environment and improve it in an orderly manner.

15. Subject Terms CIM (COLLECTION), CORPORATE INFORMATION MANAGEMENT (CIM), BUSINESS PROCESS REENGINEERING (BPR), FUNCTIONAL ECONOMIC ANALYSIS (FEA), AND GUIDEBOOK.

Security Classification of			19. Limitation of Abstract	20. # of Pages	21. Responsible Person (Name and Telephone #)
16. Report UNCLASSIFIED	17. Abstract UNCLASSIFIE D	18. This Page UNCLASSIFIED	Unlimited		MICHAEL YOEMANS 703-604-1594

Functional Process Simulation

A Guidebook

(Second Edition)

December 1993

Prepared for:

Deputy Assistant Secretary of Defense
(Information Management)
Office of the Assistant Secretary of Defense
(Command, Control, Communications and Intelligence)
Washington, DC 20301-3040

FUNCTIONAL PROCESS SIMULATION

A GUIDEBOOK
(Second Edition)

December 1993

Prepared for:

Deputy Assistant Secretary of Defense
(Information Management)
Office of the Assistant Secretary of Defense
(Command, Control, Communications and Intelligence)
Washington, DC 20301-3040

Preface

This project requires ANSER to update the existing (April 1993) Functional Process Simulation Guidebook. The objective is to demonstrate the utility of simulation and to make general observations about the application of simulation to processes as a Corporate Information Management (CIM) improvement methodology.

This task is sponsored by the Deputy Assistant Secretary of Defense for Information Management (DASD(IM)) and is administered under the direction of the Defense Information Systems Agency (DISA) through a contract with George Mason University (GMU), contract number DCA100-91-C-0033.

Table of Contents

Table of Contents - continued

List of Figures

List of Tables

CHAPTER 1

INTRODUCTION

Scope

Corporate Information Management (CIM) is a revolutionary program aimed at streamlining operations and changing the way people work in the Department of Defense (DOD). Unlike many other budgeted programs, functional managers will be directly involved in developing and executing CIM changes. To implement the CIM program rapidly and successfully, the DOD has created a Functional Process Improvement effort to encourage a consistent application of process improvement principles across its services and agencies. It encompasses the general concepts and steps associated with business process redesign as well as activity based costing (ABC) and functional economic analysis (FEA) when considering investments. Two well-known components of FPI are activity and data modeling. Using the system description standard, IDEF (for Integrated Computer Aided Manufacturing **Def**inition), activity and data models help professionals better understand their current environment and improve that environment in an orderly manner. However, while activity and data modeling are useful in context, they also have limitations, particularly in the measurement of resource expenditure (money and time). FEA is one method that can help professionals extend their understanding of money and cost. However, time-related issues or nonlinearity (caused by feedback) must be dealt with differently. Process simulation is a method that can be useful in addressing these two issues.

Overview

The purpose of this guidebook is to consider simulation (also known as dynamic modeling) as a tool for injecting temporal issues into activity modeling. The guidebook limits discussions to simulation as it relates to FPI and avoids promoting specific tools. On the other hand, it identifies criteria that can be used to evaluate simulations to perform specific tasks and cites specific tools that meet these criteria. The reader is cautioned not to draw conclusions about tools from these citing. The guidebook does not attempt to provide a comprehensive description of products, only a framework within which they can be assessed.

This chapter reviews elements of FPI and process modeling to provide a foundation for discussions on simulation. Later chapters build the case for simulation and end with a specific example.

Functional Process Improvement Review

To implement the CIM program rapidly and successfully, the DOD has used FPI to encourage a consistent application of process improvement principles across its services and agencies. It encompasses the general concepts and steps associated with business process improvement used in the civilian sector as well as the costing techniques used when considering budgets and investments.

The main focus of the CIM initiative is on management methods, and its primary objective is improvement of functions and processes. It is envisioned that future DOD system endeavors will more frequently involve the migration and evolution of assets already in place, pointing toward the development of shared data systems and software for reuse. Meeting these objectives will require the development of a functionally oriented analysis methodology that supports the concept of continuous process modernization as the new way of operating within DOD. To accelerate this process, DOD plans to take advantage of innovative FPI efforts already employed by its components and agencies. The first collection of these is the use of IDEF as the standard for activity and data modeling and ABC/FEA in handling costs and investments. A second order of efforts includes the application of simulation techniques to the IDEF activity models.

Process Modeling Review

Modeling is a common way of understanding structure and behavior of real life events. A model is a representation of reality, and provides bounds for answers to questions. However, since it is a representation and hence includes error, there is art involved in model building and use. The art is in constructing a model that is faithful to the issues to be modeled, while letting the error be in an area that is irrelevant.

IDEF provides disciplined ways of describing the structure of a system or organization. IDEF0 is a language for describing activities or processes and how they relate. Since understanding hierarchy is important in understanding large complex systems, IDEF0 is particularly useful because it includes hierarchy as an element of its modeling capability. IDEF0 supplies the structure that exists among processes and provides the framework for simulation.

IDEF1X is used for describing data requirements of a system. It defines data structure through the identification of data elements and the relationships that exist among data elements.

Simulation of data flow can be an important endeavor in understanding systems. However, for the context of this guidebook, it will be sufficient to focus on activity models and extend them into realm of simulation.

CHAPTER 2

PROCESS SIMULATION

What is Simulation?

Simulation is the use of a model to conduct experiments. The model, which changes over time, conveys an understanding of the system being represented. The purpose of experimenting using simulation is to solve problems by discovering something unknown or testing theoretical solutions to problems. The results of the experiment are then used to make prudent decisions.

Simulation is a tool that characterizes a problem, and provides a means for evaluating potential solutions. Since there are usually many possible solutions for every problem, finding potential solutions requires a thorough understanding of what constitutes the problem. This understanding is accomplished by collecting and analyzing data pertaining to the issue. Candidate solutions are then designed based upon this data and tested in the simulation environment.

Once the simulations of solutions have been conducted and the outcomes evaluated, decisions can be made. These decisions involve selecting a course of action that will have the highest probability of achieving the desired result.

Simulation proves to be an important tool in decision making. In particular,

- Simulation can promote creativity and a zest for trying new ideas.
- Simulation can predict outcomes for various courses of action.
- Simulation can account for the effects of variances occurring in a system.
- Simulation promotes total solutions.
- Simulation brings expertise, knowledge and information together.
- Simulation can be cost effective in terms of time.

Why Simulate Processes?

Most of today's systems are dynamic (as opposed to static) and stochastic (as opposed to deterministic) in nature. A dynamic system implies action. Factors that influence a system can change as time progresses. In a stochastic system these changes, such as time between failure of system components, can occur without regularity. Simulations can represent these dynamic, stochastic systems.

CHAPTER 2

PROCESS SIMULATION

What is Simulation?

Simulation is the use of a model to conduct experiments. The model, which changes over time, conveys an understanding of the system being represented. The purpose of experimenting using simulation is to solve problems by discovering something unknown or testing theoretical solutions to problems. The results of the experiment are then used to make prudent decisions.

Simulation is a tool that characterizes a problem, and provides a means for evaluating potential solutions. Since there are usually many possible solutions for every problem, finding potential solutions requires a thorough understanding of what constitutes the problem. This understanding is accomplished by collecting and analyzing data pertaining to the issue. Candidate solutions are then designed based upon this data and tested in the simulation environment.

Once the simulations of solutions have been conducted and the outcomes evaluated, decisions can be made. These decisions involve selecting a course of action that will have the highest probability of achieving the desired result.

Simulation proves to be an important tool in decision making. In particular,

- Simulation can promote creativity and a zest for trying new ideas.
- Simulation can predict outcomes for various courses of action.
- Simulation can account for the effects of variances occurring in a system.
- Simulation promotes total solutions.
- Simulation brings expertise, knowledge and information together.
- Simulation can be cost effective in terms of time.

Why Simulate Processes?

Most of today's systems are dynamic (as opposed to static) and stochastic (as opposed to deterministic) in nature. A dynamic system implies action. Factors that influence a system can change as time progresses. In a stochastic system these changes, such as time between failure of system components, can occur without regularity. Simulations can represent these dynamic, stochastic systems.

Generally, simulation addresses the dynamic properties that are often of greatest interest to management. Because an activity model is static, dynamic issues cannot be satisfactorily represented. These issues include the flow of data or other entities through an organization, contention for shared resources such as personnel or hardware, and conditional behavior of the system. Activity models are incapable of assessing flow rates, bottlenecks, idle time, throughput, cycle times, workload, and other dynamic properties. Since these dynamic properties often are of greatest interest to management, simulation becomes a key analytical tool.

As we will see in the next chapter, the analytical incompleteness of structure alone can motivate the professional to simulate. Synchronization is a challenge that may not be evident until after vast amounts of resources are invested in an organization. Also, feedback mechanisms can cause effects that defy intuition. Even when mathematical expressions capturing the qualities of feedback can be defined, these expressions may be too difficult to solve. Simulation provides a low cost means of examining such phenomena before substantial amounts of funds are invested in a project.

In addition to the above, simulation of processes is important for three specific reasons.

- First, process simulation provides a means of globally measuring changes in the output of an organization or system caused by local changes in its structure or procedures. For example, a manufacturer may find a new source for a certain component at a lower cost, but this component may fail often thus actually increasing overall cost rather than reducing it. Here, a local money saving change had the opposite global effect.

- Second, simulation's graphical presentation capabilities are useful to senior decision makers in understanding complexity through a relatively simple representation.

- Third, process simulation can identify utilization rates of activities, revealing bottlenecks or underutilization. Identified bottlenecks tell management where to apply resources or suggest a redesign of organizational architecture. Under-utilized activities tell management where waste exists. They further identify resources for re-allocation and activities that can be discontinued for a cost saving without degrading global productivity.

CHAPTER 3

IDEF0 AND SIMULATION

Viewpoint

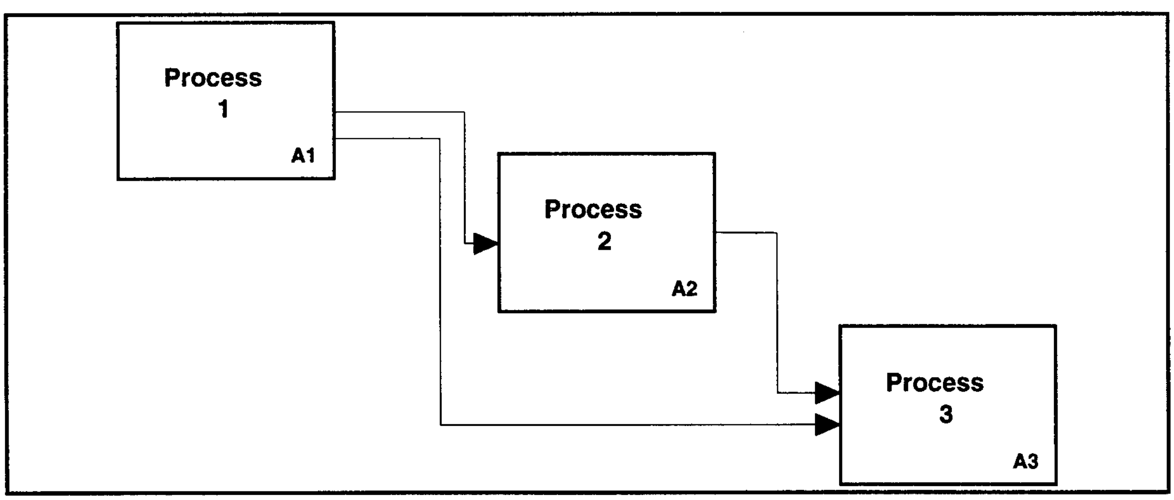
Proponents of any version of Structured Analysis and Design Technique (SADT) state that activity models first require the establishment of purpose, context, and viewpoint. However, often it is not understood that viewpoint includes a perspective in time. In a steady state context, this misunderstanding may have little impact. However, most processes modeled are not executed in a steady state. Therefore, a complete viewpoint may be difficult, if not impossible, to establish using only IDEF0 and IDEF1X descriptions. Simulation provides a suitable means of addressing this difficulty.

Process Structures

Given that the functions represented by an activity model perform dynamically, there are three generic, structural cases in an activity model that imply a need to simulate. These are graphically shown in Figures 3-1 through 3-3 below and discussed accordingly.

Synchronization

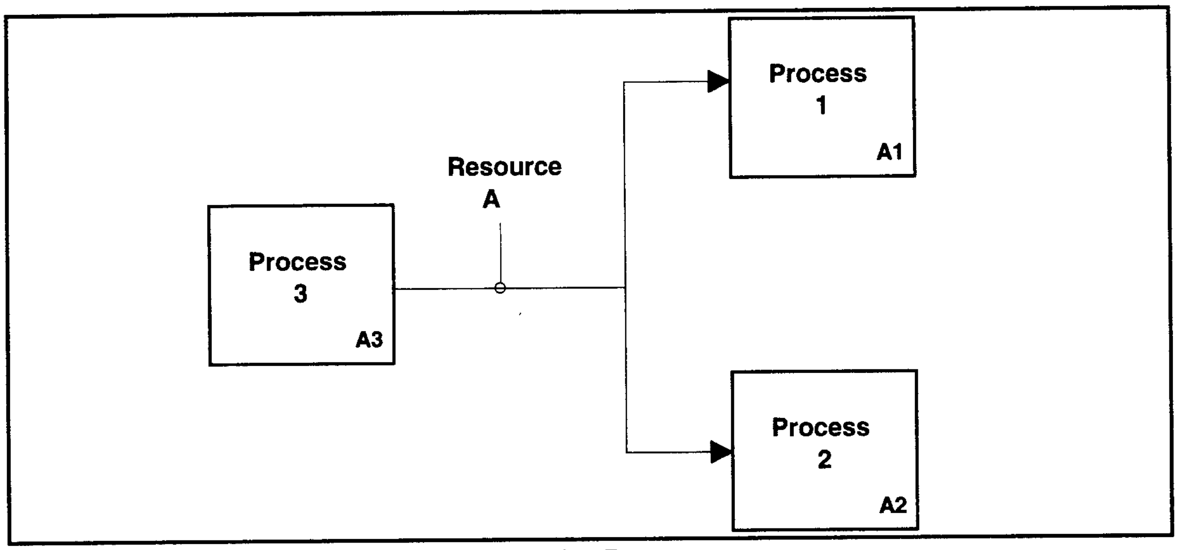
When an organization is structured so that an output from one activity provides input to two or more activities, there exists a possibility that organizational processes may "jam" because sequencing requirements are not met. Activity modeling will not reveal this deficiency, but simulation will. On the other hand, effective sequencing of activities can have dramatic impact on the overall effectiveness of an organization.



Synchronization
Figure 3-1

Conflict for Resources

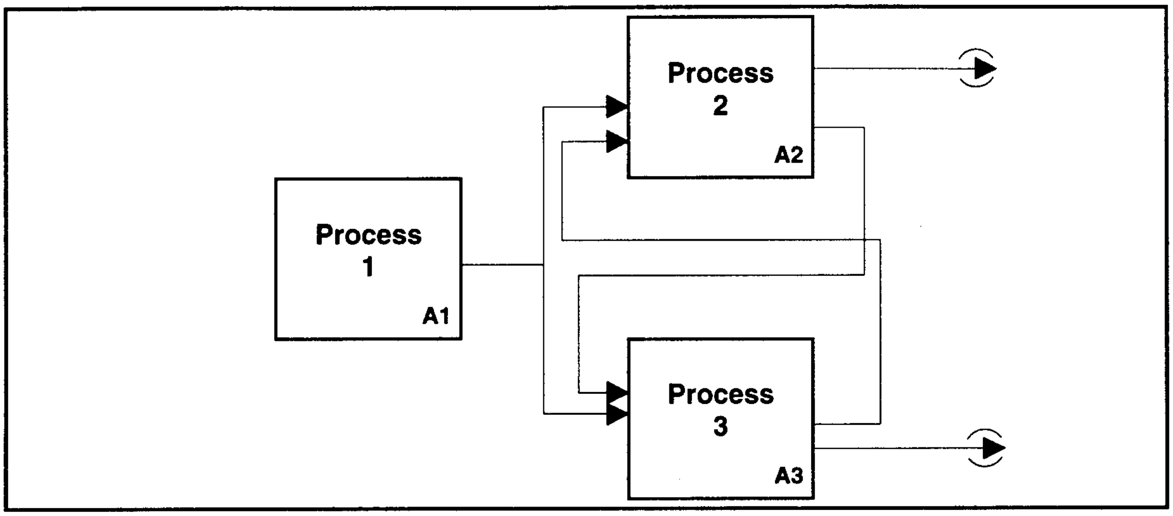
When an organization is structured so that an input to two or more activities comes from only one activity, then there exists a possibility that organizational processes may "jam" over resource conflict, particularly consumable resources. If different activities are placing demands on a common pool of resources then when the balance of resources reaches zero, one or more activities will stall until the resources are resupplied. The global impact of this effect can be marginal or substantial and simulation can quantify this effect.



Conflict for Resources
Figure 3-2

Parallelism

Finally, when outputs of two or more activities are inputs to each other, parallelism occurs. This structural situation represents coordination. Coordination is a prerequisite for process success where activities that must exchange information or other resources with other activities occurring in the same time space. Parallelism is also known as concurrency. The efficiency of information exchange between concurrent activities can have profound effects on the length of time necessary to produce a combined product.



Parallelism
Figure 3-3

Time and Motion of Resources

Simulation addresses aspects of processes for which activity and data modeling are not suited. Since activity and data models are static, they cannot cope with the impact of resource flow. Simulation is an excellent method for this purpose and, as a result, can provide insight that may be a more truthful representation of reality although in conflict with conclusions determined through other means. In the context of other process improvement methodologies, there are two general areas where simulation may uniquely contribute. These are in dynamically measuring activity utilization and system workload.

Activity Utilization

In a steady state environment, activity utilization may be measured using static techniques. One such technique is activity based costing that involves measuring both the cost and value of activities. However, in dynamic processes, while it may be possible to identify the resources supporting each activity and their sunk costs, it may be difficult to measure the proportion of these resources actually devoted to one activity. Simulation can provide these measures.

Simulation will also demonstrate bottlenecks and underutilization of activities or equipment, enabling management to redistribute resources or restructure processes to enhance overall efficiency.

System Workload

Another capability simulation provides to process improvement is in linking local changes to global measures. One aspect of activity modeling is the development of a "To-be" model. The options for projecting a "To-be" model can be unlimited. Many different collections of relationships can be redefined between activities and the application of resources used by activities can be varied in many different ways. When two or more "To-be" models are under consideration, how does one choose the best alternative? Simulation provides a means of measuring the effectiveness of the various alternatives by evaluating global performance when changes are made locally.

CHAPTER 4

SIMULATION TOOLS

General

The purpose of this chapter is to introduce the reader to types of process simulations and to discuss some simulation tools. It is not intended to be a comprehensive exposition of all tools; however, it does provide a framework for comparison.

Simulation Types

Process simulations are of two general types.

- **Discrete event simulations** replicate processes as a sequence of events where each event has a beginning point and an ending point usually measured by time. Consequently, discrete event simulations are often called time-step simulations. Associated with these discrete points in time are state variables that measure the state of the process being simulated. Therefore, as a simulation proceeds through a series of events, the process under simulation will be viewed as a series of state changes. Analysis can focus either globally or locally as designed in any particular simulation. An automobile assembly line is an example of a series of discrete events.

- **Continuous event simulations** replicate processes with mathematical expressions (often differential equations). Continuous events do not have distinct start/stop points like discrete events. Rather, the analyst can designate a sequence of points in time to provide a sequence of snapshots of the simulated process. These snapshots can be taken in global or local perspective. An example of a continuous event would be fluid flow through a system.

Process Simulation Comparison

The Table on the following page illustrates the advantages and disadvantages of each type of simulation.

Simulation Type	Advantages	Disadvantages
Discrete Simulations	• Easier to structure process simulations through discrete building blocks • Facilitates post simulation analysis keyed to initiation and completion of events • More structured framework for analysis	• Increased labor to construct simulations at discrete event level • Heavy demand on computational capability • Possible unfeasibility of the simulation due to number of events involved
Continuous Simulations	• Easier to use after appropriate choice of representative mathematical expressions • Less time to set up and for post simulation analysis • Less demand computationally	• Exactness of mathematical expressions representing the process • Possible inclusion of unacceptable errors or unknown side effects

Process Simulation Comparison
Table 4-1

Requirements for Process Simulation

- **Hierarchy.** We know through activity modeling that hierarchy is an important aspect of processes, certainly related to their complexity. A simulation needs to accommodate this hierarchical representation in terms of the decomposition of individual activities.

- **Network (Graph) Structure.** Processes are seen as networks (mathematically known as graphs). Consequently, process simulations must demonstrate this representative structure.

- **Integration Capability.** The power of activity and data modeling is in providing the required structure for analysis. Since simulation provides one such analytical tool, simulation must be an extension of the activity and data models already developed.

- **Dynamics.** The dynamic constructions discussed in the last chapter need to be accommodated. The constructions include synchronicity, resource conflict, and parallelism (or concurrency)

- **User Interface.** Software tools used to simulate processes have greater utility if data can be entered and retrieved in a way that the functional user (not just the programmer) can understand. The use of formatted windows and other graphical user interfaces (GUI) for entering data and describing chart forms are important facilities to the utility of simulation software.

- **Object orientation.** Since simulation can be difficult if developed at the code level, it is important that simulation tools be designed for application with a minimal amount (ideally none) of user coding.

- **Cost.** Simulation capability can be costly because of the expense of tools and expertise required to use them. Costs can be reduced by selection of less expensive but powerful software, use of object oriented tools, and by prudent data collection during the development of activity models.

- **Resource Attributes.** Some of the variables in a system are the individual capabilities of resources. For example, if an activity can use either a 386 or 486 PC, the duration of the activity will vary depending on which PC is used. The representation of the resource must allow for storage of these attributes.

Some Current Simulation Tools

There are several simulation tools now on the market. The list below serves to expose the reader to some of them. The tools are presented in no particular order of importance or preference.

- **Design/CPN** by Meta Software is a tool based on Petri nets.* It has been designed for integration with activity models using an automatic programming feature. Design/CPN is very flexible in the range of simulations to which it can be applied. However, the user must master the programming language ML (Meta Language) prior to using the tool. In addition, the tool contains a utility called Work Flow Analyzer that automates the translation of an IDEF0 model to a CPN simulation.

- **ProTem** is a Petri net based tool developed by Software Consultants International Limited (SCIL). It lets users model processes and procedures of varying complexity, but lacks hierarchy capability. It does allow for multiple enabling conditions for the transitions and for multiple token types in each node.

- **WITNESS** by AT&T ISTEL is a standard simulator found in the manufacturing simulation environment. WITNESS not only incorporates all of the standard simulator features, but also adds additional functionality and flexibility. Although the tool does not yet support hierarchy, it can be used for all but the most complex or application specific simulations.

- **SIMPROCESS** by CACI depicts a dynamic view of work or information flow. Although it does not support hierarchy and only a limited number of attributes can be

* Petri nets is a mathematical theory of complex systems which is particularly useful in the simulation of processes using discrete simulation. A Petri net itself is a graph consisting of interconnected locations and actions, with a set of rules governing how the occurrence of an action changes the states of the locations associated with the action.

represented. However, it is object oriented and requires no programming. It also has a report generator that can give run statistics and a cost module for rudimentary activity based costing.

- **Modeler** is a Petri net based simulation tool developed by ALPHATECH. It is object oriented and has an excellent user interface. The tool is limited in the size of systems it can model and the number of resource attributes it can represent.

- **PACE** is another Petri net based package by Grossenbacher software. The tool allows one to program complex tasks by describing them logically in an object oriented, graphical way. The package is especially useful for data flow modeling, resource management, information flow analysis, and logistical planning. Like Design/CPN, the user must master a programming language called Smalltalk. The package can be bought in modules depending upon the amount of simulation detail needed.

- **ITHINK** by High Performance Systems, Inc. is very good for modeling continuous flow. Compared to other packages it is relatively inexpensive and needs little expertise to apply. It does not, however, accommodate hierarchy or representation of attributes.

- **Custom Simulation Software** is not generally recommended for use because of the cost required to develop it. However, custom or special purpose simulations may be the best solution for replicating process dynamics if cost is not a factor. Data modeling performed in conjunction with activity modeling can define data structures that facilitate the use of standard relational data base management structures. In these cases, development costs can be reduced significantly. Custom models can be developed to feature any of the requirements described above although trade-offs should be expected.

Current and Future Trends in Simulation

- Simulation packages are being designed for PC rather than for mainframe or mini-computer use.

- Newer packages demand less programming and are more user friendly.

- More modeling is done by the system or process expert rather than the skilled programmer. This enables a direct user interface.

- Packages tend to object oriented programming where code can be reused and existing modules can be customized for particular needs.

- Graphics and animation highlight new products. This facilitates user interface and presentation of results.

CHAPTER 5

DECIDING ON SIMULATION*

Steps to a Successful Simulation

As stated previously, simulation adds depth to functional process improvement and business process redesign efforts. Generally it is not a question of if one should simulate, but rather how detailed the simulation should be. Designing a simulation requires focus and commitment. The outline below gives the steps needed for a successful simulation.

- **Define the questions to be answered by the simulation.** This is probably the most important step. One must decide what is to be achieved by the simulation. By defining the questions or problem to be solved, one can also determine the degree of detail needed in the simulation. Obviously, the more detailed the questions, the more complex the simulation needed to answer them. Vague questions will lead to vague answers and wasted time.

- **Collect activity/simulation data**. To give the simulation a firm basis, collect as much pertinent data as possible. The data can be collected through facilitated sessions, interviews, studies, and process knowledge. Accurate, up-to-date data will give the model a solid foundation and will increase the reviewer's confidence in the results. More will be said on data collection.

- **Build the activity model.** Once the data has been gathered, the activity model can then be built. The activity model is the basis upon which the simulation is constructed. During this phase, additional temporal data can be gathered if not already done so. Again, the amount of detail in the activity model, is directly related to the questions to be answered by the simulation. A detailed simulation generally requires a detailed activity model. Also one must insure the users are deeply involved in the model building. Timely user input lends accuracy to the model and lessens controversy after model completion.

- **Build the simulation.** Once the activity model has been constructed, the simulation will follow. Using the activity model as a guide, the simulation can be constructed and the runs executed based upon the timing data collected earlier.

- **Validate the simulation.** A number of runs should be done to validate the results. The results can be validated by comparing the simulation performance to actual performance of the process being simulated. Numerous runs with wide variation of parameters provide the best means for validation.

* Appendix B consolidates the contents of this chapter into a checklist for those considering simulation.

- **Experiment with the simulation.** Now comes the time to answer the questions posed in the first item of this outline. One needs to experiment with many model parameters to ensure that the questions are answered. Once again, the more runs that can be made, the better. Additionally, sensitivity runs can be conducted and alternative "To-Be" configurations can be tested.

- **Analyze the results.** When satisfied that the simulation answers the desired questions, the results can be analyzed and the conclusions drawn. These analyzed results are then presented to the decision makers for guidance and further action.

- **Present the results.** To implement changes to improve processes, the simulation results must be presented to the decision maker. A well thought out simulation, itself, may be the best sales tool. However, some simulation packages do not lend themselves to briefing decision makers. In that case, a clear, concise presentation of results obtained and recommendations should be presented.

Selecting a Simulation Package

The previous chapter discussed specific simulation packages now on the market. This section, based on an excerpt from JMI Consulting's, *Improve Quality and Productivity with Simulation*, will give thoughts on how to choose the right simulation tool.

- **Current and future applications.** One of the first items to be addressed is what are the current and future simulation needs of an organization. The selected package's capabilities should be as wide as the number of potential application areas. If the processes are solely manufacturing oriented, then a package that specifically addresses manufacturing operations will suffice. If a wide variety of systems are to be modeled, then a more flexible simulation tool is needed.

- **Level of detail.** The level of detail needed in the simulation must correspond directly to the detailing capability of the tool. For example, if a simulation is used for modeling complex command and control systems, then a package that provides intricate detailing and hierarchy should be used. Additionally, in processes describing rates of flow, continuous event modeling may be necessary. In some cases both discrete and continuous event modeling may be necessary. Many packages do not possess the capability to do both.

- **Time constraints.** Projects subject to time constraints such as short completion schedules will likely need a tool with rapid model building capabilities and little or no programming necessary. Building models rapidly requires graphical interfaces, good debugging capability, and rapid execution times.

- **Simulation builders and end users.** Tool proficiency time is another consideration. Simulation builders with little or no programming experience will need a tool which

minimizes code writing, such as one heavily object oriented. This also applies to the end users of the package. The end users must be able to interpret the results and, if necessary, interact with the model. Both the simulation builder and user must evaluate the tool's documentation, training requirements and customer support. Each of these factors can influence the "spin-up" time needed to adequately use the package.

- **Simulation size limitations.** The size of the simulation grows as elements and detail are added to it. A complex system being simulated may require many elements and interrelationships to adequately portray it. Some simulators are fixed in size or must be bought in modular building blocks. As an example, a manufacturing simulator may have a fixed number of machine and labor types. In addition, model size affects execution time. It follows that larger models will require more time to run. Testing of several alternatives will also require multiple runs, all adding to the total simulation time.

- **Operating environment.** The operating environment (PC, Mac, VAX, etc.) varies for different packages. Some operate only in one, while others have or will have multiple versions. Simulation builders must closely look at their current environment prior to choosing a package. Also, a Local Area Network (LAN) version will be useful if there are multiple users of the simulation. A LAN version will prevent log-jams at a single access point.

- **Statistical capabilities.** These can be viewed in two ways. One involves the packages ability to collect statistical data during the runs and to display that data. The second deals with what standard statistical distributions are included with the package. A good selection of distributions, representative of the empirical data, is a necessity for a simulation package. Finding distributions of empirical data by hand is tedious and a waste of analysis time.

- **Global attributes.** Attributes are a modeling feature that allow an object to be identified (sometimes uniquely) by a number of characteristics. For example, a car could be represented by its name, model and color. Attributes remain with an object as it travels through a process, but it is desirable to be able to change attribute values at any given point in the model.

- **Import/export of data.** Input data for simulations is sometimes extracted from large data bases and it is helpful to be able to download this data to a file that can be used with the simulation package. Likewise, the ability to write output from the simulation to a data file is useful, especially if statistical analysis must be done outside the simulation package.

- **Human intervention in model runs.** Performing multiple runs can be very time consuming. Thus it can be advantageous to make runs during nonworking hours. Packages that allow multiple runs without human intervention can save time. This feature should be considered for large models when lengthy run times are expected.

Simulation Data

Data is the substance of simulation and modeling and, as such, deserves special attention. This section will examine the aspects of data collection and data analysis

Data Collection Procedures and Measures

Structured Analysis and Design Technique (SADT) and other methods used in the development of IDEF models are structured data collection techniques that can be modified to include data required for simulation. Using the IDEF0 as a baseline, there are some specific questions that should be asked during interviews, facilitation sessions, and document review.

- **Cycles*.** What defines a cycle for a process? The answer to this question generally results from the IDEF0 diagrams.

- **Cycle Frequency.** How many cycles of each subprocess are required for each cycle of the parent process? Again, the IDEF0 diagrams can give insight into this question. Study of the IDEF0 activity decompositions will reveal cycles of subprocesses. It is paramount that IDEF0 modelers fully describe the cycles in the activity narratives.

- **Cycle Duration (Time).** What is the time required for each cycle? IDEF0 diagrams will not directly give the answer as they are time independent. The temporal issues must be gleaned from questions to the subject matter experts who are assisting in IDEF0 model construction.

- **Resources (Cost and Value).** What resources are consumed and produced by each cycle of the process? What are their costs and values? Resources consumed and produced are usually the inputs and outputs of the IDEF0 diagrams. These are readily recognizable. Values and costs of these resources can be determined by costing measures such as activity based costing (ABC).

- **Controls.** What controls a cycle? That is, what information or resources are required to initiate and terminate a cycle? What additional information or resources are required to sustain a cycle between initiation and termination? The IDEF0 diagrams will identify the control measures and activity inputs. Subject matter experts will provide what additional information and/or resources are required.

* Cycles form the temporal basis of simulations since they define the time span of a process from beginning to end. Processes will have subprocesses which, in turn, will have subcycles.

Data Analysis

Data analysis depends on the purpose of the simulation effort. That is, what data one chooses to analyze is dependent upon the questions the simulation is to answer. Nonetheless, the design of such analysis can be guided by some general principles of system theory.

Basic measures of performance for information systems include availability, utilization, throughput, response time, workload and system balance. These measures can apply to the whole system, segments of the system, subsystems, or components of the subsystems.

- **Availability** is the ratio of the number of times the system, segment, etc. is ready for use to the total number of times it is needed.
- **Utilization** is the ratio of time in use to time available for use.
- **Throughput** is the level of work done over a period of time.
- **Response time** is the time elapsed from the end of an input to the start of a response to that input.
- **Workload** is the number of inputs or the number of demands per unit of time.
- **System balance** is the distribution of idle, busy, and blocked segments, subsystems and components during particular periods of time.

Combinations of the above can also prove useful for analysis.

- Availability versus throughput
- Throughput versus workload
- Utilization versus workload
- Response time versus workload

CHAPTER 6

LESSONS LEARNED FROM PREVIOUS SIMULATION EFFORTS

The following are some items collected from past simulation tasks. They should prove of assistance for future efforts.

- Simulation provides a temporal perspective to process modeling.

- Simulation can help quantify costs and benefits of processes. This capability can support functional economic analysis.

- Simulation provides a means of measuring the value of improvements expected from "To-Be" models. Ideas can be tried out without huge expenditures of funds.

- Process simulation validates activity and data modeling.

- If IDEF models are built with simulation in mind, the need for very clear explanations of the logic inside the activities is paramount. Inputs, outputs, controls, and mechanisms (ICOMS) must be carefully identified.

- Simulation can reveal utilization rates of activities and the global value of local changes.

- The most important factor in simulations is to have a clear idea of what questions the simulation is to answer. Most systems are complex and have many variables. Without a clear view of the end goal, it is impossible to restrict the model to only the relevant dimensions and therefore the problem will remain too complex and nothing will be resolved.

- The questions that need to be answered influence not only the simulation itself, but also the modeling tool chosen. The more complex the questions, the more powerful the tool needed.

- Data for simulations should be collected during the activity and data modeling phases.

  -- Data collection for simulation forces activity and data modelers to understand processes better.

  -- The activity and data modeling includes collection of data about processes. Inclusion of timing data saves follow-on collection efforts.

- Generic data required for simulations include

  -- Procedures for defining a process cycle, including rules for initiation, control and termination of a cycle

  -- Time required for each cycle

  -- Number of cycles required of child cycles for the parent cycle

  -- Quantity and value of resources consumed or used by a process

  -- Conditional behavior rules involving the effect on cycle time or output values of choosing one resource over another.

CHAPTER 7

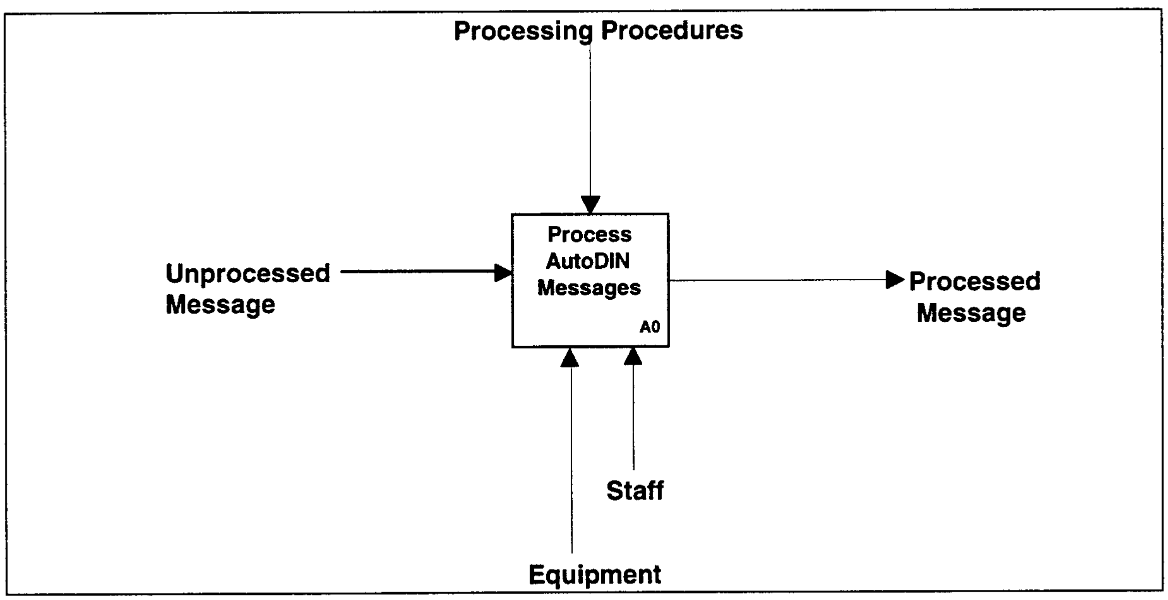
CASE EXAMPLE: A MESSAGE PROCESSING SYSTEM

Background

Activity at a message processing center (MPC) during a military exercise was chosen as a case study for process simulation. The MPC selected was on the Automated Digital Information Network (AutoDIN) system supporting a NATO Southern Region War exercise (WINTEX/CIMEX) in Spain. Messages, generated by the headquarters are processed at the MPC to be entered into AutoDIN and transmitted to other nodes.

Objective

The problem for analysis was a rapidly generating backlog of messages and unacceptably increasing waiting times for transmission. A desired solution would reduce these waiting times and associated backlogs. Message backlog is a common problem of command and control systems which surfaces during exercises and operations where message traffic is significantly higher than during routine activities.

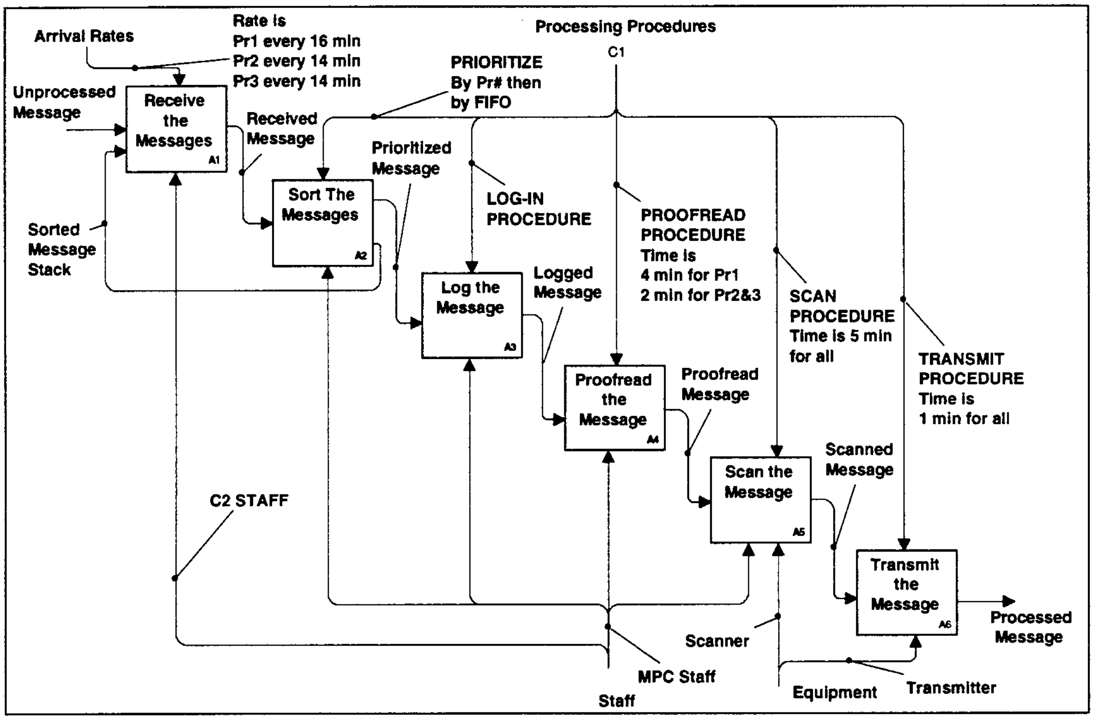


A-0: Message Processing Center
Figure 7-1

Details of the MPC Process

There were three types of messages that arrived at the center: immediate, priority, and status (referred to as priority one, two, and three, respectively). An operator would select a message, giving precedence to priority one messages, review it for errors, and

hand feed it into an optical scanner. Once a message was scanned, it was automatically transmitted and the operator was free to select a new message. Priority one messages arrived approximately every sixteen minutes, while priority two and priority three messages arrived approximately every fourteen minutes. Priority one messages took four minutes (on the average) to log in and review, while priority two and priority three messages took two minutes to log in and review. The scanning process took five minutes (on the average) to complete regardless of the priority level. Transmission took an average of one minute regardless of the priority level. The first two levels of an IDEF0 activity model of the functions of the MPC are shown in Figures 7-1 and 7-2 respectively.



A0: Process and Transmit Messages
Figure 7-2

Mechanisms for the A2, A3, and A5 Activities shown in Figure 7-2 reveal options for reducing the backlog of messages. Four cases were developed from the two general options shown below.

- Add manpower to the preprocessing of messages (checking for correct format and priority) and structure the use of manpower.

- Improve the scanning capability by replacing the existing scanner with a more responsive scanner or additional scanners.

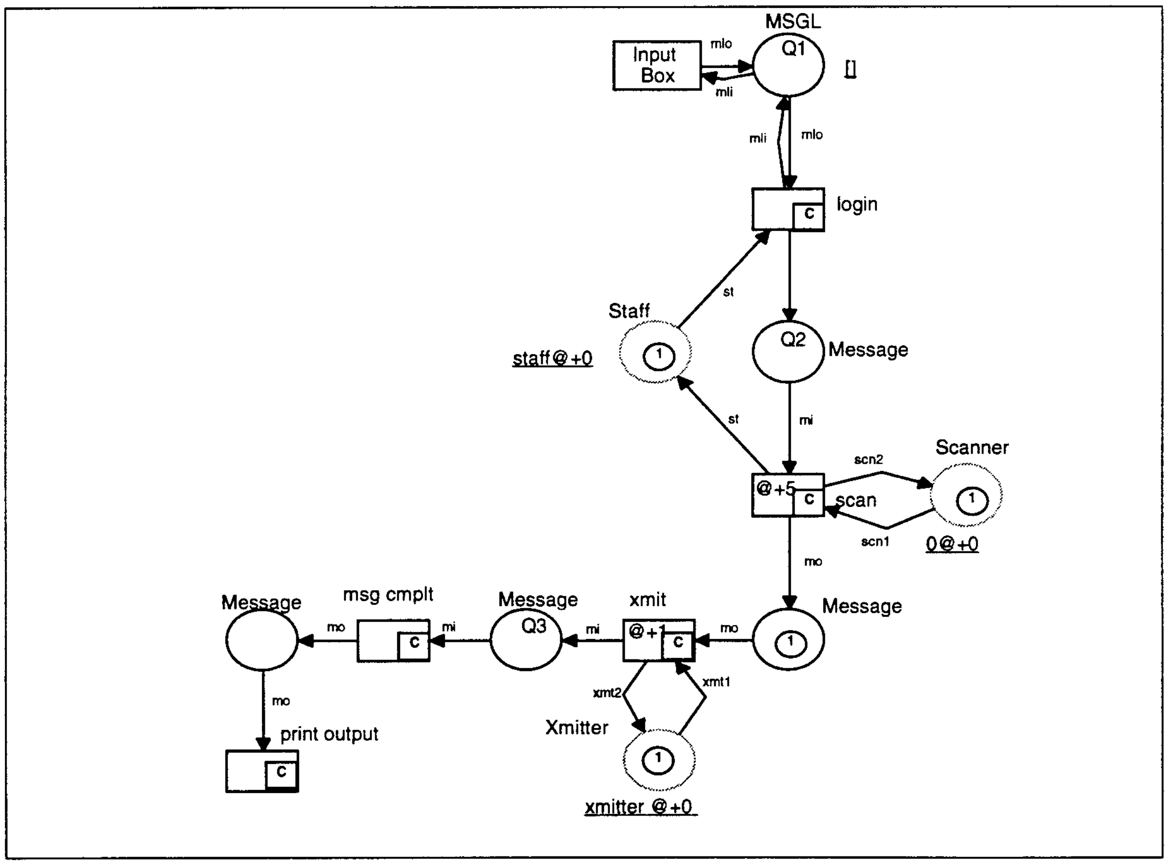
A conclusion reached by the MPC supervisor before simulation was introduced was that the message bottleneck was caused by a slow scanner (an average of 5 minutes per message, see Activity A5). This conclusion was based only on his subjective

assessment. However, a member of his team suggested that the backlog would be reduced if more manpower was added to the process. A simulation was developed on this hypothesis.

Simulation Methods, Analysis and Results

Methods

Design/CPN was used to model the AutoDIN message processing center. This model is shown in Figure 7-3. It shows "arcs" (path of messages) through nodes of two types. The circle node is called a "place" representing a queue of messages (e.g., buffer) for an activity. The rectangle node is called a "transition" representing the activity. (In activity models, boxes combine the representation of queues and activities.) As intuition would suggest, every transition is preceded by at least one place and feed only into places. Similarly, places feed only into transitions.



Message Processing Center
Simulation Diagram
Figure 7-3

Petri net tokens which flow through the network path were used to model the resources and associated attributes. Resources modeled included messages and staff. The attributes modeled for messages included

- Priority type
- Arrival time to the system
- Exist time from the system
- Other time statistics.

We have omitted some technical description as to exactly how transitions work and how attributes are assigned; however, it is sufficient to note that the simulation is constructed largely in a lego-like fashion where the building blocks are tokens (resources), places (queues), transitions (activities), and arcs (relationships between the activities).

In this model, as messages arrive at the center they are sorted according to priority. This is done with an appropriate code segment, executed each time the transition **Input Box** is enabled. (The term enabled means that all necessary conditions have been satisfied for the activity to begin.) Messages then wait at place **Q1** until the transition **login** is enabled (there is at least one message at **Q1** and one operator at the place Staff). When **login** is enabled, an output token will be created at place **Q2**. The arc inscription adds two or four to the token's time stamp, depending on priority level. The message waits until the transition scan is enabled (there is at least one message at **Q2** and at least one token at the place scanner representing that the scanner is available). When **scan** is enabled, a token is created at the output place with an added time delay of five, representing the five minutes needed to scan. Similarly, **xmit** will be enabled if there is at least one message at its two input places. A time delay of one is added to the output token (one minute is needed to transmit a message). The message then exits the system. Appropriate code segments were also added to keep the time statistics associated with each message. Output was sent to a separate file to be processed. However, line graphs can be built within Design/CPN.

Simulation Analysis

- **Case 1 - The Baseline, one staff member to log, proofread, and scan messages:** For all four cases 296 messages (24 hours of message flow) were simulated. A backlog of 117 (about 40%) messages was generated for this 24 hour simulated period of time. Only one of the priority one messages and 13 of the priority two messages were in the backlog. Most of the backlog (103 messages) consisted of priority three messages*.

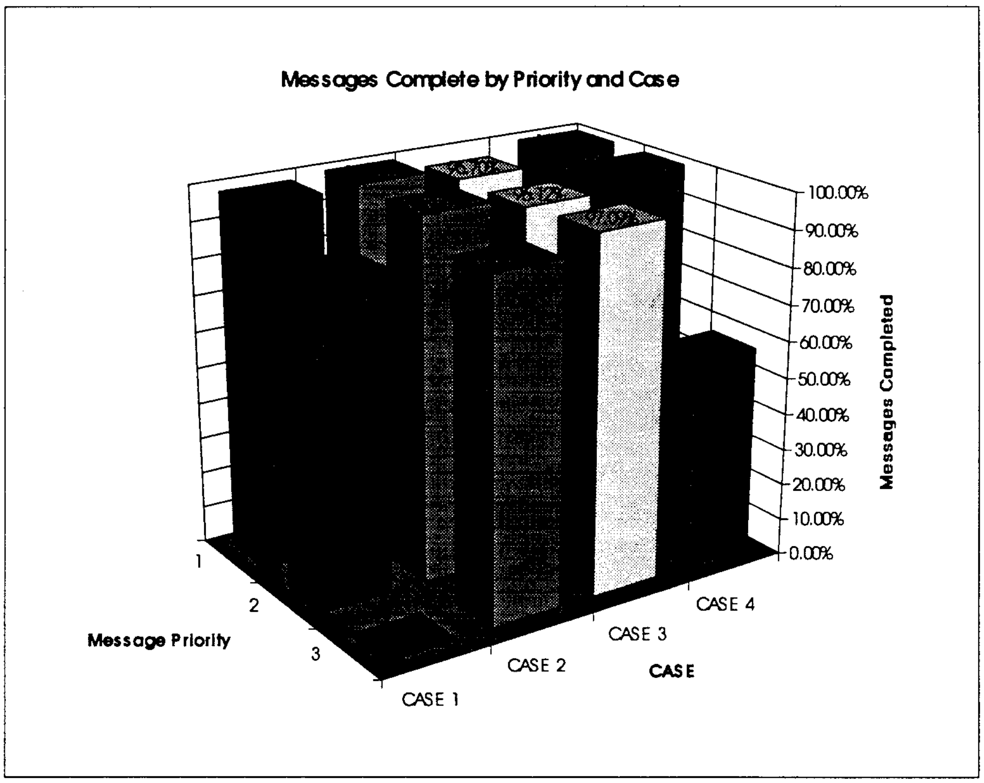
---

* It should be noted that these results validated the model as they matched the real world results of the exercise.

- **Case 2 - Two staff members to log, proofread, and scan messages:** In this case, the backlog was reduced dramatically to a total of 9 (no priority one, one priority two, and eight priority three messages).

- **Case 3 - Two staff members, one to log and proofread, and one to scan:** In this case the backlog was reduced to a total of 10 but with an increase in priority one messages to three. Priority two messages were reduced to four and priority three messages were reduced to three.

- **Case 4 - Baseline modified by reducing scanning time to 3 minutes:** The backlog was reduced to 44 priority three messages.



Message Processing Center
Simulation Results By Case and Message Priority
Figure 7-4

Results

While the supervisor was correct in assessing that replacement of the scanner with one having improved performance characteristics would reduce the message backlog (117 to 44 in the case simulated), clearly a better option for reducing the backlog was to add a staff member, preferably as was done in Case 2 above.

Observations

Two considerations not evaluated in the simulation above are the cost of adding manpower or technology and the value of improved performance. Had these factors been incorporated, we would have the remaining parameters for conducting a functional economic analysis.

Cost can be a relatively easy to compute from a salary schedule. However if seen as an opportunity cost, computation could be more difficult. Opportunity cost is the lost value caused by moving the required manpower from another staff position. Manpower is a scarce resource in military organizations, causing the choice of using manpower to be a fundamental economic decision.

Determining value can be difficult. With an MPC, it is important to understand how messages are used. For instance, what activities will be delayed as a result of not receiving a message within a certain time window and what is the impact of the delay? To determine the answer, it may be appropriate to extend the simulation above to include activities requiring message flow.

APPENDIX A

THEORETICAL PARADIGMS FOR SIMULATING PROCESSES*

Introduction

In developing process simulations, paradigms+ forming a conceptual framework for dynamic process descriptions can be useful for reference. The use of paradigms in simulation gives the analyst some understanding of the scope of analysis and the requirement for analytical capabilities or tools needed. Three paradigms that can be used are queuing, state-transition, and cybernetic. Each of these paradigms has its own set of assumptions and measures, and each presents a distinct though limited view of a process. However, combined they can provide a comprehensive view. The three paradigms are described below.

Process Paradigms

Queuing

The queuing paradigm views an organization of processes as a service system in which "customers" (e.g. messages, logistical commodities, and other consumable resources) arrive at a service facility (activity) to be serviced by "servers" (mechanisms such as information systems, personnel teams, etc.). In queuing customers must wait for an activity to be processed. A queuing system exhibits two types of behavior, transient (over a short time period) and steady-state (over a long time period), and is specified completely by seven characteristics.

- Input or arrival distribution
- Output or departure distribution
- Number of servers
- Service discipline (e.g. first in - first out (FIFO), last in - first out (LIFO), etc.)
- Maximum number of customers allowed in the system

* This appendix is directed at the analyst and is technical. It may be skipped with no overall loss in understanding of the simulation of processes.

+ The paradigms were developed from the study of command and control systems which are particularly complex. However, the observations generally apply to most information systems.

- Maximum waiting time in queue distribution
- Calling source (the population that generates arrivals)

The primary objective of a process organization as seen from a queuing perspective is to reduce the customers waiting time for activities. The types of measures that can be derived from the queuing paradigm are the expected waiting time per customer and expected number of customers in a specific queue or in the entire system. Also, for systems with a maximum waiting time in the queue, the probability of reneging from the queue can be determined ("Reneging" is leaving the queue without being serviced.)

These measures imply that the queuing paradigm addresses throughput issues such as how long did it take, how many were serviced, how many were missed, how often was each server occupied, and where were the bottlenecks?

State-Transition

The State-Transition paradigm views process organization (system) as a random process in which the system is characterized at discrete points in time by a set of random variables, each representing the organization's state at each a designated time step. This random process can be viewed as goal directed in that the system is trying to reach a final "desired state" (e.g. having everyone in the organization notified at a specified time).

The basic assumption of the State-Transition paradigm is that the state of the system at time t(k) is dependent only on the state of the system at t(k-1). Consequently, an important variable is the probability that the system will transition into state X(k) at time t(k) given that it is in state X(k-1) at time t(k-1).

The State-Transition paradigm is not concerned with how long it takes the system to transition from one state to the next (it allows for the possibility that the state remains unchanged from one time step to the next) but is concerned with the probability that a state change will occur. Therefore, the primary objective of an organizational process as seen from a State-Transition perspective is to maximize the probability that the system reaches the final goal state. The type of process issue addressed with the State-Transition paradigm is how a change in the state transition probabilities at time t(k) affects the probability that the system will achieve its final goal state.

A major drawback to the State-Transition paradigm is identifying all the state variables needed to accurately describe a system or organization at any given time. Given the complexity of today's organizations and systems, the magnitude of the number of variables may computationally limit the implementation of this paradigm. However, it has been successfully applied where an aggregation of state variables is possible or where the behavior of the analyzed system is particularly sensitive to a finite subset of parameters.

Cybernetic Paradigm (The Adaptive Control Cycle)

The cybernetic paradigm views information systems as a goal-directed adaptive control cycle process with feedback. The information system consists of a control agent (e.g. headquarters) and objects of control (resources). Information is passed from the control agent to the objects of control and status information is passed, in turn, from the objects of control back to the control agent. The cybernetic paradigm is a functional perspective in that a set of functions is specified that describes the internal workings of the control agent (e.g. gathering information about the situation, assessing the situation, deciding what to do about it, planning the implementation of the decision, disseminating the plan and monitoring its execution). Inherent in these functions is a decision process by which status information (feedback) is converted in command information. The outputs of this conversion are tasks assigned to the objects of control (i.e. subordinates), thus giving purpose (i.e. a goal) to the entire system.

A characteristic of the cybernetic paradigm is that it focuses on describing what goes on inside the control agent. However, there seem to be no uniform set of measures associated with this paradigm because it incorporates several disciplines (e.g. elements of optimal control theory, estimation theory, and decision theory) which have not yet been integrated into a single theory of information management. Notwithstanding this perspective, we will later isolate a set of measures which when used with certain simulation applications will capture critical elements of this paradigm.

The primary objective of a command and control system from a cybernetic perspective is to maintain the object of control within "desired" bounds. For example, a commander may give a subordinate unit a set of objectives in which he wants to maintain at least a 3-to-2 force ratio in his favor, otherwise, steps will be taken to reinforce the subordinate unit. The control agent (headquarters) will receive feedback from the environment that determines the current force ratio. If the ratio is above the "desired" ratio then the system is under control (i.e. the plan is working), however, if the ratio is below this level, then the system is out of control and appropriate actions must be taken (e.g. adjust the current plan or develop a new plan).

Selecting a Paradigm

The paradigms above are neither mutually exclusive nor all inclusive. For example, the concept of a system's state at a given time is inherent in all three. Therefore, in choosing a paradigm, it is not a matter of which paradigm is better, but which paradigm provides a perspective that is consistent with what you want to know about the information system. For example, if the time it takes an information system to convert status information into command information is an important issue, then a queuing perspective with its probability of state transition measures would be appropriate. Analyzing complex systems, such as command and control, requires viewing the system from more than one perspective.

APPENDIX B

SIMULATION CHECKLIST

STEP 1 Should you simulate?		
Question	**Yes**	**No**
Is time an important issue in your activity models?		
Will there be multiple "To-Be" models?		
Are large investments in equipment contemplated for the "To-Be" configuration?		
Are synchronization, conflict for resources or parallelism important issues or prominent from the activity diagrams?		
Do you anticipate changes to subprocesses that will have great effect on the main processes?		
Are resources consumed erratically?		
Are resource flow bottlenecks suspected?		

STEP 2 Assuming that you should simulate, carefully develop the questions you want answered by the simulation.

STEP 3 Select a simulation tool
Determine your current and future simulation needs.
Determine the level of detail needed from the simulation.
Determine time constraints for the project.
Determine the simulation builders and end users.
Determine size of the simulation.
Determine the operating environment.
Determine the package's statistical capabilities.
Determine the attributes of the various objects.
Determine the data import/export needs.
Determine the amount of human intervention needed to run the package.

STEP 4 Collect data
Define cycles.
Determine cycle frequency.
Measure cycle duration.
Determine resources used, consumed, and produced.
Identify cycle controls.

STEP 5 Build the activity model

STEP 6 Build the simulation

STEP 7 Validate the simulation

STEP 8 Experiment with the simulation
Experiment with model parameters.
Conduct sensitivity analysis.
Run "To-Be" scenarios.

STEP 9 Analyze results
Availability
Utilization
Throughput
Response time
Workload
System balance

STEP 10 Present the results

BIBLIOGRAPHY

James A. Brimson, *Activity Accounting, An Activity Based Costing Approach*, The Wiley/National Association of Accountants Professional Bank Series, John Wiley & Sons, Inc., New York, 1991.

Anne J. Clough, Charles Stark, Draper Laboratory, "Choosing an Appropriate Modeling Technology," *Cross Talk*, August 1992.

Colonel Gary Q. Coe, USA, "Measuring Force Multiplication," *Proceedings of the 1988 Symposium on Command and Control Research*, pp 63-69. Sponsored by Kazie Research Group Technical Panel on C3, Joint Directors of Laboratories and National Defense University and in cooperation with LEGG Control Systems Society, September 1988.

Colonel Gary Q. Coe, USA, and Dr. John T. Dockery, "OJCS Initiatives in C2 Analysis and Simulation," *Science of Command and Control, Coping with Uncertainty*, pp 19-31, AFCEA International Press, Washington, DC, 1988.

Colonel Gary Q. Coe, USA, "The Search for Uncertainty in an Uncertain World," Professional Papers of AFCEA's 9th Western Conference, Anaheim, CA, January 1988.

Department of Defense Manual, DOD 8020.1M (Draft), "Functional Process Improvement (Functional Management Process for Implementing the Information Management Program of the Department of Defense)," Director of Defense Information, Office of the Secretary of Defense, January 1993.

Thomas J. Gogg and Jack R. A. Mott, *Improve Quality and Productivity with Simulation*, JMI Consulting Group, 1992.

IDEF User's Group, *User's Manual, ICAM Definition (IDEF) Method for Information Modeling (IDEF1X)*, IDEFUG-M-1X-8511, reproduced with the approval of WL/MTI (Air Force Wright Aeronautical Laboratories), Nov 1985.

IDEF User's Group, *User's Manual, ICAM Definition (IDEF) Method for Function Modeling (IDEF0)*, IDEFUG-M-0-8106, reproduced with the approval of WL/MTI (Air Force Wright Aeronautical Laboratories), Nov 1985.

Kurt Jensen, Soren Christiansen, Peter Huber, and Meers Holla, "Design CPN, A Reference Manual, Version 1.75," published by Meta Software, August 1991.

David A. Marca and Clement L. McGowan, *SADT Structured Analysis and Design Technique*, McGraw-Hill Book Company, 1988.

Meta Software Corporation, "Design/CPN Tutorial Version 2.0 for the Macintosh," April 1992.

J. L. Peterson, *Petri Net Theory and the Modeling of Systems*, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1981.

Robin Ray, Janice Snow, and Barbara Dooley, *User's Manual, Design IDEF for the Apple Macintosh*, published by Meta Software Corporation, Cambridge, MA, 1992.

Paul Strassman, *The Business Value of Computers, An Executive's Guide*, The Information Economics Press, New Canaan, CT, 1990.

Åke Wilström, *Functional Programming Using Standard ML*, Prentice Hall International Series in Computer Science, London, 1987.